Uncertainty-Aware Self-training for Low-Resource Neural Sequence Labeling

Jianing Wang[1], **Chengyu Wang**[2], **Jun Huang**[2], **Ming Gao**[1,3*], **Aoying Zhou**[1]

[1] School of Data Science and Engineering, East China Normal University, Shanghai, China
[2] Alibaba Group, Hangzhou, China
[3] KLATASDS-MOE, School of Statistics, East China Normal University, Shanghai, China
lygwjn@gmail.com, {chengyu.wcy, huangjun.hj}@alibaba-inc.com, {mgao, ayzhou}@dase.ecnu.edu.cn

Abstract

Neural sequence labeling (NSL) aims at assigning labels for input language tokens, which covers a broad range of applications, such as named entity recognition (NER) and slot filling, etc. However, the satisfying results achieved by traditional supervised-based approaches heavily depend on the large amounts of human annotation data, which may not be feasible in real-world scenarios due to data privacy and computation efficiency issues. This paper presents SeqUST, a novel uncertain-aware self-training framework for NSL to address the labeled data scarcity issue and to effectively utilize unlabeled data. Specifically, we incorporate Monte Carlo (MC) dropout in Bayesian neural network (BNN) to perform uncertainty estimation at the token level and then select reliable language tokens from unlabeled data based on the model confidence and certainty. A well-designed masked sequence labeling task with a noise-robust loss supports robust training, which aims to suppress the problem of noisy pseudo labels. In addition, we develop a Gaussian-based consistency regularization technique to further improve the model robustness on Gaussian-distributed perturbed representations. This effectively alleviates the over-fitting dilemma originating from pseudo-labeled augmented data. Extensive experiments over six benchmarks demonstrate that our SeqUST framework effectively improves the performance of self-training, and consistently outperforms strong baselines by a large margin in low-resource scenarios.

Introduction

Neural sequence labeling (NSL) is one of the fundamental tasks in natural language processing (NLP) with a broad range of applications, including named entity recognition (NER) (Li et al. 2022) and slot filling (Liu et al. 2022), which aims at classifying language tokens into a pre-defined set of classes (Shen et al. 2021; Agarwal 2022). Previous works have achieved satisfying performance by developing well-designed deep architectures and/or fine-tuning pre-trained language models (PLMs) (Ma and Hovy 2016; Li et al. 2020; Zhou, Li, and Li 2022; Zhang et al. 2022). Yet, these approaches heavily depend on massive labeled data, which could be even more bothersome in some real-world scenar-

ios, such as over-fitting on limited labeled data and privacy constraints.

Recently, a branch of the semi-supervised learning (SSL) paradigm (Chawla and Karakoulas 2005; van Engelen and Hoos 2020; Yang et al. 2021) arises to bypass the aforementioned issues, which aims to utilize effectively large-scale unlabeled data in addition to the few-shot labeled data. Self-training is one of the typical SSL techniques which can be viewed as teacher-student training (Grandvalet and Bengio 2004; Amini et al. 2022). Concretely, a *teacher* model is trained over the labeled data and then be used to generate pseudo labels on the unlabeled data. After that, a *student* model can be optimized by the augmented data, and be used to initialize a new *teacher*.

Benefit from self-training, previous methods make a remarkable success on a series of *instance-level classification* tasks, such as image classification (Zhou et al. 2021; Wang et al. 2022a; Liu et al. 2021) and text classification (Meng et al. 2020; Mukherjee and Awadallah 2020; Yu et al. 2021; Hu and Khan 2021; Tsai, Lin, and Fu 2022; Kim, Son, and Han 2022). In contrast to instance-level classification, we observe that there are two challenges in applying standard self-training to NSL. On one hand, the task of NSL is based on the *token-level classification*, which requires the model to capture the inherent token-wise label dependency. On the other hand, the teacher model inevitably generates some noisy labels that cause error accumulation (Wang et al. 2021a). Some sample selection strategies (e.g., model confidence, uncertainty estimation) and consistency regularization mitigate the effect of noisy labels and alleviate the problem of confirmation bias (Do, Tran, and Venkatesh 2021; Cao et al. 2021; Rizve et al. 2021; Wang et al. 2021b; Andersen and Maalej 2022). However, it is unclear how these methods can be applied to token-level classification.

To remedy this dilemma, we develop SeqUST, a novel semi-supervised learning framework for NSL, which improves standard self-training via two decomposed processes, i.e., reliable token selection and robust learning. Specifically, we first pseudo annotate the unlabeled data. Then, a Monte Carlo (MC) dropout technique, which is the approximation technique in Bayesian Neural Network (BNN) (Gal and Ghahramani 2016; Wang and Yeung 2016), is used to estimate the uncertainty of each language token derived from the teacher model. We judiciously select the reliable tokens

from each unlabeled sentence based on the model confidence and certainty. Finally, we introduce two training objectives to improve the model robustness, i.e., Masked Sequence Labeling (MSL) and Gaussian-based Consistency Regularization (GCR). In MSL, we generate a masked matrix to make the model only focus on the selected reliable tokens in each sentence. We also utilize partially huberised cross-entropy (PHCE) loss to explicitly mitigate the effect of label noises. In GCR, we assume each selected token embedding follows the Gaussian distribution and perturbs token embeddings to encourage consistency between perturbed embeddings and original representations. This technique effectively reduces the risk of over-fitting.

We perform extensive experiments over multiple benchmarks with very few labeled data (10/20/50/100 examples per class). We adopt BERT-base (Devlin et al. 2019) as our backbone. Results show that our SeqUST outperforms strong baselines by a large margin in low-resource settings.

In a nutshell, we make the following main contributions:

- We develop a novel semi-supervised neural sequence labeling framework SeqUST to alleviate the problem of data scarcity in low-resource scenarios.

- We propose the token-level uncertainty estimation to empower self-training. Furthermore, two training objectives are introduced to improve the model robustness, including Masked Sequence Labeling and Gaussian-based Consistency Regularization.

- Extensive experiments over multiple benchmark datasets demonstrate that our proposed framework achieves substantial performance improvement.

Related Work

Semi-supervised Learning and Self-training

SSL aims to utilize effectively unlabeled data in addition to labeled data, which has been widely used in the NLP community (Yang et al. 2017; Gururangan et al. 2019; Xie et al. 2020; Chen, Yang, and Yang 2020). For instance, Yang et al. (2017); Gururangan et al. (2019) utilize variational autoencoders (VAEs) for sequence classification and labeling. Unsupervised data augmentation (UDA) (Xie et al. 2020) generates augmented data by back translation, and leverages consistency regularization for unlabeled data. Chen, Yang, and Yang (2020) proposes MixText to mix labeled, unlabeled and augmented data, and performs similar consistency training as UDA.

Self-training is one of the earliest methods in SSL that has recently shown state-of-the-art performances (Chen, Yang, and Yang 2020; Meng et al. 2020; Li, Li, and Ouyang 2021). Recent works improve standard self-training by considering sample selection strategies, including model confidence (Bengio et al. 2009; Kumar, Packer, and Koller 2010) and uncertainty estimation (Cao et al. 2021; Tsai, Lin, and Fu 2022). For example, Cao et al. (2021) presents uncertainty-aware self-training (UST) to sample pseudo-labeled data by BNN. (Tsai, Lin, and Fu 2022) propose CEST which aims to leverage graph-based contrast induction to solve the problem of confirmation bias and data

smoothness on the selected data. However, they focus on instance-level classification, it would be highly desirable if they can be applied to token-level classification.

Low-resource Neural Sequence Labeling

Low-resource NSL aims at classifying the input language tokens with very few labeled data. Prior researches address this problem via meta-learning (Ziyadi et al. 2020; Ma et al. 2022) and augmentation learning (Ding et al. 2020; Zhou et al. 2022; Wang et al. 2022b). For instance, (Ziyadi et al. 2020) leverages prototypical network (Snell, Swersky, and Zemel 2017) to learn adaptable knowledge from few-shot episode data. Wang et al. (2022b) and Zhou et al. (2022) aim to generate the in-domain augmented data through masked language modeling (MLM) (Devlin et al. 2019).

Yet, these methods ignore the informative semantics of unlabeled data. To reach this goal, a series of previous works (Miller, Guinness, and Zamanian 2004; Peters et al. 2017) focus on token representations enhancement by pre-training word embeddings on unlabeled data. Another line of research focuses on latent variable modeling (Chen et al. 2018), adversarial training method SeqVAT (Chen et al. 2020) and cross-view training method CVT (Clark et al. 2018). Recently, MetaST (Wang et al. 2021a) attempts to apply self-training to NSL by two techniques, i.e., adaptive validation set construction by uncertainty estimation, and noisy data re-weighting via student loss. In contrast, we perform reliable token selection by the joint estimations of the teacher confidence and uncertainty, and two well-designed training objectives support robust training. Compared with previous methods, we achieve the best overall performance with very few labeled data.

Preliminaries

We first present the notations and then introduce the background knowledge of the Bayesian neural network.

Notations

We represent a labeled dataset and an unlabeled dataset as $\mathcal{D}_l = \{X_i, Y_i\}_{i=1}^{N_l}$ and $\mathcal{D}_u = \{X_i\}_{i=1}^{N_u}$, respectively. $X_i = \{x_{ij}\}_{j=1}^{L}$ denotes the input sentence with L language tokens. $Y_i = \{y_{ij}\}_{j=1}^{L}$ is the label sequence and $y_{ij} \in \mathcal{Y}$ is the tag of token $x_{ij} \in \mathcal{X}$, $\mathcal{X}$ and $\mathcal{Y}$ denote the token vocabulary of the PLM and the label space, respectively. N_l and N_u represent the numbers of labeled and unlabeled data, respectively ($N_l \ll N_u$). The goal of semi-supervised NSL is to learn a neural mapping function $f^W : \mathcal{X} \to \mathcal{Y}$ over labeled data $\mathcal{D}_l$ and unlabeled data $\mathcal{D}_u$, where W is the collection of model parameters.

Bayesian Neural Network (BNN)

Similar to CEST (Tsai, Lin, and Fu 2022), in this part we briefly describe BNN. Given a neural model f^W, the vanilla BNN assumes a prior distribution over its model parameters W. In other words, BNN averages over all the possible weights instead of directly optimizing for the weights (Mukherjee and Awadallah 2020). Given a labeled dataset $\mathcal{D}_l$, the parameter can be optimized by the posterior
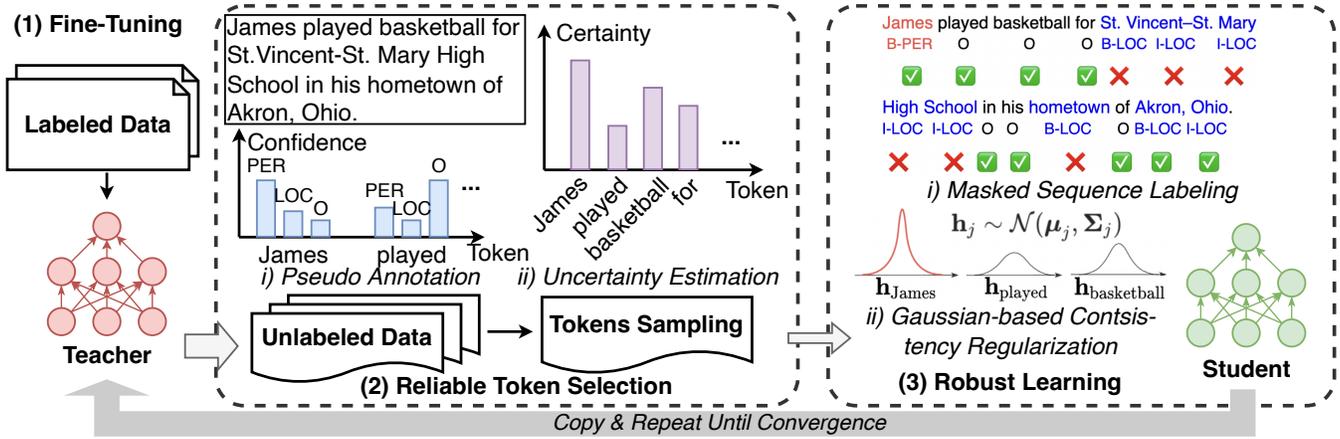
Figure 1: The framework overview. 1) We first fine-tune a teacher model over the labeled data. Then, 2) we utilize MC dropout to perform uncertainty estimation and select reliable tokens for each unlabeled sentence. 3) Two novel training objectives are proposed to improve the model robustness and alleviate the over-fitting issue. (Best viewed in color.)
.

distribution $p(W|\mathcal{D}_l)$. During model inference, given one unlabeled example $X_i \in \mathcal{D}_u$, the probability distribution of each token $x_{ij} \in X_i$ for class $c \in \mathcal{Y}$ is $p(y = c|x_{ij}, X_i) = \int_W p(y = c|f^W(x_{ij}, X_i)p(W|D_u)dW$. Yet, it is intractable in practice for calculation. To make it tractable, we can find a surrogate tractable distribution $q_\theta(W)$ that makes the model posterior easy to compute. Thus, we consider $q_\theta(W)$ to be the dropout distribution (Srivastava et al. 2014) which aims to sample T masked model weights $\{\widetilde{W}_t\}_{t=1}^T \sim q_\theta(W)$ from the current model. The approximate posterior for each token is:

$$p(y = c|x_{ij}, X_i) \approx \frac{1}{T} \sum_{t=1}^T p(y = c|f^{\widetilde{W}_t}(x_{ij}), X_i). \quad (1)$$

Methodology

In this section, we propose the SeqUST framework to improve the self-training paradigm for low-resource NSL. The framework overview is illustrated in Fig 1.

Pseudo Annotation

In the initial stage, a vanilla PLM (e.g., BERT) can be fine-tuned over the labeled dataset $\mathcal{D}_l$ to form a *teacher* model f^{W^*}, where W^* is the collection of parameters. The hard label $\tilde{y}_{ij}$ of each token x_{ij} in the given unlabeled sentence $X_i \in \mathcal{D}_u$ can be pseudo annotated by the teacher model f^{W^*}. Formally, we have:

$$\tilde{y}_{ij} = \arg\max_c p(y = c|f^{W^*}(x_{ij}), X_i), \quad (2)$$

where $p(\cdot)$ is the probability distribution, which can be modeled as a softmax classifier or a conditional random field (CRF) (Lafferty, McCallum, and Pereira 2001) layer.

Reliable Token Selection

Prior works (Mukherjee and Awadallah 2020; Tsai, Lin, and Fu 2022; Rizve et al. 2021) improve self-training by

the instance-level selection strategy, which aims to utilize BNN to perform uncertainty estimation for each example, and then select the reliable examples from the whole unlabeled dataset that the model is most certain about. Different from them, we focus on the token-level uncertainty estimation, and aim to select reliable tokens from each sentence.

Token-level Uncertainty Estimation. We assume that each sentence is independent of another and can be measured individually. Specifically, we follow (Houlsby et al. 2011; Gal, Islam, and Ghahramani 2017) to leverage information gain of the model parameters to estimate how certain the model is to the pseudo-labeled tokens with respect to the true labels. Formally, given one input sentence $X_i \in \mathcal{D}_u$, we have:

$$\mathbb{B}(\tilde{y}_{ij}, W|x_{ij}, \mathcal{D}_u) = \mathbb{H}(\tilde{y}_{ij}|x_{ij}, \mathcal{D}_u) - \\ \mathbb{E}_{p(W|\mathcal{D}_u)}[\mathbb{H}(\tilde{y}_{ij}|x_{ij}, W)], \quad (3)$$

where $\mathbb{H}(\cdot)$ is the entropy function, $x_{ij} \in X_i$ and $\tilde{y}_{ij} \in \widetilde{Y}_i$ denote the token and tag, respectively. $\mathbb{B}(\tilde{y}_{ij}, W|x_{ij}, \mathcal{D}_u)$ denotes the information gain which is the difference between $\mathbb{H}(\tilde{y}_{ij}|x_{ij}, \mathcal{D}_u)$ (the final entropy after seeing all tokens from unlabeled sentences) and $\mathbb{H}(\tilde{y}_{ij}|x_{ij}, W)$ (the current entropy for the token x_{ij}). $p(W|\mathcal{D}_u)$ is the posterior distribution. As the calculation of Eq. 3 is intractable, we utilize MC dropout in BNN to perform approximation. Specifically, we assume that the posterior distribution $p(W|\mathcal{D}_u)$ can be replaced with dropout distribution $q_\theta(W)$. Thus, we can sample T masked model weight $\{\widetilde{W}_t\}_{t=1}^T \sim q_\theta(W)$, and calculate the approximation value of $\mathbb{B}$ as:

$$\hat{\mathbb{B}}(\tilde{y}_{ij}, W|x_{ij}, \mathcal{D}_u) = -\sum_{c \in \mathcal{Y}} (\frac{1}{T} \sum_{t=1}^T \hat{p}_c^t) \log(\frac{1}{T} \sum_{t=1}^T \hat{p}_c^t) \\ + \frac{1}{T} \sum_{t=1}^T \sum_{c \in \mathcal{Y}} \hat{p}_c^t \log(\hat{p}_c^t), \quad (4)$$

where $\hat{p}_c^t = p(\tilde{y}_{ij} = c|f^{\widetilde{W}_t}(x_{ij}), X_i)$ is the predict probability for the token x_{ij} derived from the t-th masked model.

Thus, a lower $\hat{\mathbb{B}}(\tilde{y}_{ij}, W | x_{ij}, \mathcal{D}_u)$ value means that the model is more certain about the prediction, as higher certainty corresponds to lower information gain.

Tokens Sampling. For reliable token selection, we jointly consider the confidence and certainty value. For the confidence of the prediction $\tilde{y}_{ij}$, we have:

$$s_{ij}^{cf} = \frac{1}{T} \sum_{t=1}^{T} p(y = \tilde{y}_{ij} | f^{\widetilde{W}_t}(x_{ij}), X_i). \quad (5)$$

A higher confidence value s_{ij}^{cf} means the model is more confident for the pseudo label $\tilde{y}_{ij}$. Theoretically, selecting language tokens with high confidence predictions moves decision boundaries to low-density regions, which is satisfied with the low-density assumption (Rizve et al. 2021). However, many of these selected tokens with higher confidence are incorrect due to the poor calibration of neural networks (Guo et al. 2017), which brings the conformation bias problem. To reduce the wrong labels, we additionally design a certainty score s_{ij}^{ct} based on the uncertainty estimation as:

$$s_{ij}^{ct} = 1 - \hat{\mathbb{B}}(\tilde{y}_{ij}, W | x_{ij}.\mathcal{D}_u), \quad (6)$$

Intuitively, if the model is always certain about some tokens, these tokens might be too easy to contribute any additional information. To this end, we can obtain the final sampling weight for each token as:

$$s_{ij} = \frac{s_{ij}^{cf} \times s_{ij}^{ct}}{\sum_{x_{ij} \in X_i} s_{ij}^{cf} \times s_{ij}^{ct}}. \quad (7)$$

Robust Learning for NSL

Masked Sequence Labeling (MSL). With the measure of model confidence and certainty, we can use them to sample reliable tokens for each sentence. However, each token has the location and label dependency constraint, we can not directly remove the tokens that are not sampled. Thus, we define a masked matrix to record the sampling results, i.e.,

$$M_{i,j} = \begin{cases} 1 & x_{ij} \text{ has been sampled;} \\ 0 & x_{ij} \text{ has not been sampled;} \end{cases} \quad (8)$$

when $M_{ij} = 0$, it means the corresponding label may be a noise and should be masked during self-training.

Generally, we can define the following cross-entropy function as the training loss:

$$l(X_i, \widetilde{Y}_i, M_i) = \frac{1}{L_i'} \sum_{j=1}^{L_i'} \mathbb{I}(M_{ij} = 1) \log p_W(x_{ij}, \tilde{y}_{ij}), \quad (9)$$

where $p_W(x_{ij}, \tilde{y}_{ij}) = p(y = \tilde{y}_{ij} | f^W(x_{ij}), X_i)$ is the prediction probability derived from the student model, $L_i' = \sum_{j=1}^{L} \mathbb{I}(M_{ij} = 1)$ is the number of the selected tokens in X_i. $\mathbb{I}(\cdot)$ denotes the indicator function. However, it is still possible that the selected tokens could be wrong pseudo annotated although the token with higher confidence and certainty. To explicitly mitigate the effect of label noises, we follow (Tsai, Lin, and Fu 2022) to utilize partially huberised cross-entropy loss (PHCE loss) as our noise-robust

---

Algorithm 1: Self-training Procedure of SeqUST

Require: Neural model f^{W_0}, labeled data $\mathcal{D}_l$, unlabeled data $\mathcal{D}_u$.
1: Initialize a teacher model $f^{W^*} = f^{W_0}$;
2: **while** not converged **do**
3:      Fine-tune the teacher model f^{W^*} over the labeled data $\mathcal{D}_l$;
4:      Pseudo annotate each unlabeled sentence $X_i \in \mathcal{D}_u$ by Eq. 2
        to obtain the hard labels $\widetilde{Y}_i$;
5:      Initialize a student model $f^W = f^{W_0}$;
6:      **for** $X_i \subseteq \mathcal{D}_u$ **do**
7:         Obtain the confidence score s_{ij}^{cf} for $x_{ij} \in X_i$ by Eq. 5;
8:         Obtain the certainty score s_{ij}^{ct} for $x_{ij} \in X_i$ by Eq. 6;
9:         Sample reliable tokens by the sampling weight in Eq. 7,
           and generate a masked matrix M_i;
10:        Calculate the training loss $l(X_i, \widetilde{Y}_i, M_i)$ in Eq. 10;
11:        Calculate the regularization loss $R(f^W, X_i)$ in Eq. 14;
12:        Update the model f^W by reduce $\mathcal{L}(W)$ in Eq. 15;
13:      **end for**
14:      Update the teacher model $f^{W^*} = f^W$;
15: **end while**
16: **return** The teacher model f^{W^*}.

---

loss, which is based on a simple variant of gradient clipping for the classification loss, e.g. cross-entropy. Hence, the loss function in Eq. 9 can be modified as:

$$l(X_i, \widetilde{Y}_i, M_i) = \frac{1}{L_i'} \sum_{j=1}^{L_i'} \mathbb{I}(M_{ij} = 1) \phi(x_{ij}, \tilde{y}_{ij}), \quad (10)$$

where

$$\phi(x, y) = \begin{cases} -\tau p_W(x, y) + \log \tau + 1 & p_W(x, y) \leq 1/\tau; \\ -\log p_W(x, y) & p_W(x, y) > 1/\tau; \end{cases} \quad (11)$$

is the PHCE loss function, $\tau > 1$ is the hyper-parameter. Thus, the model learned by Eq. 11 can be more robust to the noisy labeled tokens than the common cross-entropy.

Gaussian-based Consistency Regularization (GCR). During iterative self-training, it is possible that the model biases the sampling process toward picking *easier* tokens, which have higher confidence and certainty. This inevitably leads to the student model over-fitting on these frequently selected samples. Previous methods (Chen, Yang, and Yang 2020; Xie et al. 2020) solve this problem by utilizing back translation (Sennrich, Haddow, and Birch 2016) to generate instance-level augmented data, which aims to translate each sentence into a different language and then translate it back. However, it can not be directly applied to token-level task (Lee et al. 2021). To reach this goal, we elaborately design Gaussian-based consistency regularization (GCR) to perturb token embeddings in the hidden space and make the model prediction consistent.

Given one pseudo-labeled sentence $X_i = \{x_{ij}\}_{j=1}^{L}$, we can obtain the corresponding contextual representations $\mathbf{H}_i = \{\mathbf{h}_{ij}\}_{j=1}^{L}$ derived from the final hidden layer output of the student model (i.e., BERT). $\mathbf{h}_{ij} \in \mathbb{R}^h$ denotes the representations of the token x_{ij}, where h is the hidden size. We assume that token embeddings follow Gaussian distributions (Das et al. 2022; Lee et al. 2021), i.e.,

$\mathbf{h}_{ij} \sim \mathcal{N}(\boldsymbol{\mu}_{ij}, \boldsymbol{\Sigma}_{ij})$. Specifically, we use two projection network $\mathcal{F}_{\mu}(\cdot)$ and $\mathcal{F}_{\Sigma}(\cdot)$ with ReLU activation to produce Gaussian distribution $\boldsymbol{\mu}_{ij}$ and $\boldsymbol{\Sigma}_{ij}$ for each token x_{ij}. Formally, we have:

$$\boldsymbol{\mu}_{ij} = \mathcal{F}_{\mu}(\mathbf{h}_{ij}), \boldsymbol{\Sigma}_{ij} = \mathcal{F}_{\Sigma}(\mathbf{h}_{ij}), \tag{12}$$

where $\boldsymbol{\mu}_{ij} \in \mathbb{R}^h$, $\boldsymbol{\Sigma}_{ij} \in \mathbb{R}^h$ represent mean and diagonal covariance of the Gaussian embedding, respectively. We leverage the reparameterization trick (Kingma and Welling 2014) to perturb the input x_{ij} with the sampled noise without altering its semantics. Specifically, we generate K perturbed representations $\{\hat{\mathbf{h}}_{ij}^{(k)}\}_{k=1}^{K}$, where

$$\hat{\mathbf{h}}_{ij}^{(k)} = \mathbf{h}_{ij} \odot (\boldsymbol{\mu}_{ij} + \boldsymbol{\Sigma}_{ij} \odot \epsilon^{(k)}). \tag{13}$$

Here, $\epsilon^{(k)} \sim \mathcal{N}(\mathbf{0}, \mathbf{I}_h)$, and $\mathbf{0} \in \mathbb{R}^h$, $\mathbf{I}_h \in \mathbb{R}^h$ are the vector with zeros and identity matrix, respectively. $\odot$ denotes the element-wise multiplication. Afterward, the KL divergence objective can be used to control the probability distribution consistency between the original semantic representation and the perturbations for each token of x_{ij}:

$$R(f^W, X_i) = \frac{1}{L \cdot K} \sum_{j=1}^{L} \sum_{k=1}^{K} D_{\text{KL}}\big(p_W(y|\mathbf{h}_{ij}) || p_W(y|\hat{\mathbf{h}}_{ij}^{(k)})\big) \tag{14}$$

where $D_{\text{KL}}(\cdot || \cdot)$ denotes the KL divergence function, and $p_W(y|\cdot)$ is the probability distribution derived from the student model. Finally, we update the student model by modifying the following objective:

$$\mathcal{L}(W) = \sum_{X_i \in \mathcal{D}_u} \big(l(X_i, \widetilde{Y}_i, M_i) + \lambda R(f^W, X_i)\big), \tag{15}$$

where λ is the pre-defined hyper-parameter balancing the regularization term.

Self-training Procedure

The training algorithm of our SeqUST framework is shown in Algorithm 1. Specifically, at each iteration stage, we fine-tune a teacher model f^{W^*} on the few-shot labeled data $\mathcal{D}_l$, and pseudo annotate the unlabeled data $\mathcal{D}_u$ (Algorithm 1, Line 3-4). Then, we select reliable tokens in each sentence by the model confidence and certainty (Algorithm 1, Line 7-9), and use them to update the parameters of the student model f^W by modifying the final loss $\mathcal{L}(W)$ (Algorithm 1, Line 5, 10-12). Finally, the trained student model f^W is used to initialize a new teacher f^{W^*}, and repeat the above steps till convergence.

Experiments

Datasets and Implementation Details

We choose six widely used benchmarks to evaluate our SeqUST framework, including SNIPS [1] (Coucke et al. 2018) and Multiwoz [2] (Budzianowski et al. 2018) for slot filing,

Datasets	Domain	Type	#Class	#Train	#Test
SNIPS	Dialogue	Slot Filling	54	13.6k	0.7k
Multiwoz	Dialogue	Slot Filling	15	20.3k	2.8k
Movie	Review	NER	13	7.8k	2.0k
Restaurant	Review	NER	9	7.7k	1.5k
CoNLL-03	News	NER	5	14.0k	3.5k
OntoNotes	General	NER	19	60.0k	8.3k

Table 1: The statistics of each dataset.

MIT Movie [3] (Liu et al. 2013b), MIT Restaurant [4] (Liu et al. 2013a), CoNLL-03 (Sang and Meulder 2003) and OntoNotes (Weischedel et al. 2013) for NER. The statistics of each dataset are shown in Table 1. For each dataset, we use a greedy-based sampling strategy to randomly select 10-shot labeled data per class for the few-shot labeled training set and validation set, while the remaining data are viewed as unlabeled data.

During self-training, the teacher and student model share the same model architecture. In default, we choose BERT-base-uncased (Devlin et al. 2019) from HuggingFace[5] with a softmax layer as the base encoder. We use grid search to search the hyper-parameters. We select five different random seeds for the dataset split and training settings among $\{12, 21, 42, 87, 100\}$. We report the averaged F1 scores with the standard deviation on the whole testing set. We implement our framework in Pytorch 1.8 and use NVIDIA V100 GPUs for experiments.

Baselines

We adopt several state-of-the-art semi-supervised methods as our strong baselines [6]. **Standard self-training (SST)** (Huang et al. 2021) is the simple method that generates hard pseudo labels and uses them to train a student model. **VAT** (Miyato et al. 2019) and **SeqVAT** (Chen et al. 2020) utilize adversarial training method with consistency learning to improve the robustness. **CVT** (Clark et al. 2018) is based on cross-view training for semi-supervised sequence labeling. **MetaST** (Wang et al. 2021a) aims to select reliable validation data and trains the student model by re-weighting strategy. We also choose the standard **Fine-tuning** as the supervised learning-based baselines. We reproduce their results with the same settings.

Main Results

Table 2 illustrates the main results of our framework SeqUST compared with other baselines. The results of fine-tuning over full data are the ceiling performance. With only 10 labeled training data per class, we achieve the best averaged F1 score of 77.78%. In addition, SeqUST outperforms the few-shot fine-tuning and standard self-training by 7.99% and 6.36%, respectively. The vanilla fine-tuning

---

[1] https://github.com/sonos/nlu-benchmark/tree/master/2017-06-custom-intent-engines.

[2] https://github.com/budzianowski/multiwoz.

[3] https://groups.csail.mit.edu/sls/downloads/movie/.

[4] https://groups.csail.mit.edu/sls/downloads/restaurant/.

[5] https://huggingface.co/transformers.

[6] We do not consider N-way K-shot few-shot NSL baselines because they have different learning settings with ours.

Baselines	SNIPS	Multiwoz	Movie	Restaurant	CoNLL-03	OntoNotes	Avg.	Δ
Full Data								
Fine-tuning	97.00	88.00	87.30	79.00	91.90	89.20	88.73	-
Few Labeled Data (10-shot)								
Fine-tuning	79.10±0.38	71.40±0.25	69.50±0.40	53.75±0.19	71.82±0.28	73.15±0.25	69.79	+0.00
Few Labeled Data (10-shot) + Unlabeled Data								
SST	81.07±0.40	72.25±0.11	71.14±0.22	55.13±0.29	72.86±0.34	75.07±0.18	71.25	+1.46
VAT	79.70±0.22	72.08±0.30	68.80±0.39	54.26±0.33	72.60±0.33	73.38±0.27	70.14	+0.35
SeqVAT	79.62±0.41	72.17±0.34	68.83±0.35	54.02±0.40	73.18±0.28	73.71±0.30	70.23	+0.44
CVT	79.22±0.44	71.42±0.38	69.03±0.51	54.17±0.42	71.70±0.72	73.88±0.40	69.90	+0.11
MetaST	<u>86.74±0.33</u>	<u>77.34±0.51</u>	<u>77.52±0.39</u>	<u>63.02±0.29</u>	<u>76.88±0.41</u>	<u>77.69±0.24</u>	<u>76.53</u>	<u>+6.74</u>
SeqUST	**87.33±0.30**	**77.98±0.26**	**77.73±0.22**	**64.19±0.36**	**79.10±0.27**	**80.33±0.44**	**77.78**	**+7.99**

Table 2: The performance comparison of F1 scores (%) with standard deviations on six benchmarks. Δ denotes an improvement over the few-shot fine-tuning method compared to our framework. All models (except fine-tuning with full data) are trained with 10-shot labeled samples for each class and overall F1 aggregated over five different runs with different random seeds.

Models	SNIPS	Movie	CONLL-03
SST	81.07±0.40	72.25±0.11	72.86±0.34
SeqUST	**87.33±0.30**	**77.73±0.22**	**79.10±0.27**
w/o. selection	82.37±0.39	70.12±0.29	73.40 ±0.37
w/o. confidence	86.98±0.33	76.99±0.29	77.60 ±0.29
w/o. certainty	83.33±0.30	72.29±0.33	73.16±0.31
w/o. PHCE	87.01±0.34	77.32±0.27	78.31±0.35
w/o. GCR	87.11±0.31	77.25±0.23	77.93±0.31

Table 3: The 10-shot ablation study (F1 score %).

#-shot⟶	SeqUST			MetaST		
	20	50	100	20	50	100
SNIPS	92.13	93.44	95.60	91.99	92.82	95.10
Multiwoz	79.70	82.05	83.36	79.45	81.34	84.27
Movie	80.80	83.16	84.98	80.29	82.75	84.35
Restaurant	69.02	73.95	75.70	67.93	72.83	75.28
CoNLL-03	81.74	83.20	85.59	78.54	82.34	85.10
OntoNotes	82.26	84.00	85.77	80.78	82.44	85.01

Table 4: The F1 score (%) with different numbers (20/50/100 examples per class) of labeled data.

Selection Strategy	SNIPS	MIT Movie	CONLL-03
None	45.75	51.77	49.88
Confidence	43.18	48.06	45.22
Certainty	20.42	27.60	29.10
All	**17.76**	**23.10**	**24.63**

Table 5: The error rate (%) of different selection strategy. None means does not select tokens.

method without any unlabeled data performs worst than expected since the number of labeled data is extremely insufficient for the parameter modification. The performance of the self-training-based approaches (including SST, MetaST, and SeqUST) is consistently better than others, which indicates the effectiveness of the self-training paradigm. MetaST is a strong baseline that performs a token-level re-weighting strategy in self-training. Yet, MetaST ignores the robust learning of some noisy labels even though they have higher weights, and does not consider the consistency regularization, its performance can be considered sub-optimal in contrast with ours. Results show that SeqUST achieves high improvement over the state-of-the-art MetaST by 1.25%.

Ablation Study

In Table 3, we randomly choose three datasets to conduct an ablation study to investigate the characteristics of the main components in SeqUST. The results show that no matter which module is removed, the model performance is affected. All variants still outperform standard self-training, even though removing some components. When removing the reliable token selection (w/o. selection), the performance declines a lot because that many noisy labels hurt the model effectiveness. We also observe that the result of w/o. certainty is lower than w/o. confidence, which demonstrates

that the model uncertainty is more useful for label denoising, and greatly alleviate the conformation bias issue. Moreover, the use of robust loss (i.e., the PHCE loss in MSL) and consistency regularization consistently contribute to the robustness improvement when training on pseudo-labeled data.

Further Analyses

Data Efficiency. We further explore the model effects with different numbers of labeled data per class (#-shot) among $\{20, 50, 100\}$. Results in Table 4 illustrate that the performance gradually improves as the number of labeled data increases, as expected. In addition, we also find that our SeqUST consistently outperforms the strong baseline MetaST no matter how many labeled training examples. This can be attributed to the introduction of uncertainty-aware self-training with well-designed robust learning approaches.

Unlabeled Text	**Peter Weller plays** a reassembled indestructible **cop** in this **futuristic cop movie directed by Paul Verhoeven**
Pseudo Labels	B-Actor I-Actor O O O B-Plot I-Plot O O B-Opinion B-Genre I-Genre O O B-Director I-Director
Ground Truth	B-Actor I-Actor O O B-Plot I-Plot I-Plot O O B-Opinion B-Genre O O O B-Director I-Director

Unlabeled Text	It **quoted** an **Interior Ministry** statement as **saying Shabir Ahmad Muhammad** Jalil was **executed in Mecca**
Pseudo Labels	O O O B-ORG I-ORG O O O B-PER I-PER I-PER I-PER O O O B-LOC
Ground Truth	O O O B-ORG I-ORG O O O B-PER I-PER I-PER I-PER O O O B-LOC

Figure 2: Cases of pseudo-labeled texts. The language tokens in bold are sampled as reliable tokens.
.
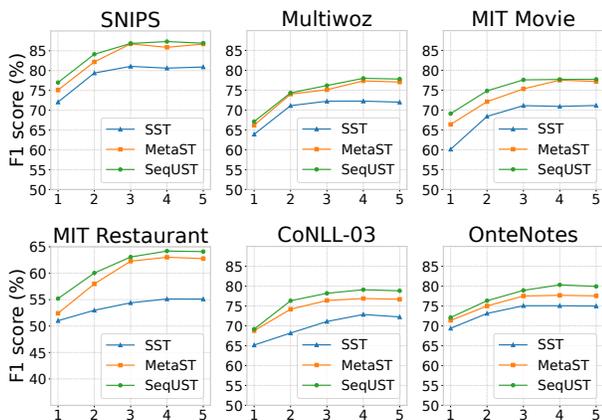
Figure 3: The performance (F1 score %) of different self-training iterations over six benchmarks.
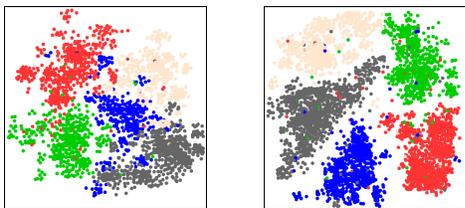
Figure 4: The t-SNE visualization of SeqUST w/o. GCR (left) and w/ GCR (right). (Best viewed in color.)

Effectiveness of Reliable Token Selection. To demonstrate the effectiveness of reliable token selection, we choose three datasets to investigate the error rate on the unlabeled data. As shown in Table 5, we obtain the lowest error rate with both token-level confidence and certainty, indicating the merit of discarding noisy pseudo labels. In addition, we respectively choose two cases from MIT Movie and CONLL-03 to show the pseudo labels and selected tokens. The results in Fig. 2 show that 1) most tokens can be correctly predicted and 2) our selection strategy can identify some noisy labels.

Effectiveness of the Robust Learning. We first demonstrate the importance of PHCE loss for robust learning in the masked sequence labeling task. As shown in Fig. 3, our framework consistently outperforms the standard self-training by a large margin, which does not use robust learning methods. Moreover, the performance of MetaST is not

Datasets	BERT		BiLSTM	
	SoftMax	CRF	SoftMax	CRF
SNIPS	87.33±0.30	86.95±0.25	84.94±0.20	85.16±0.28
Multiwoz	77.98±0.26	78.20±0.38	73.13±0.19	73.92±0.20
Movie	77.73±0.22	78.02±0.24	72.05±0.26	73.12±0.31
Restaurant	64.19±0.36	66.17±0.26	60.83±0.42	61.11±0.47
CoNLL-03	79.10±0.27	78.21±0.35	75.53±0.22	75.22±0.23
OntoNotes	80.33±0.44	79.74±0.40	76.02±0.34	76.28±0.34
Avg.	77.78	77.88	73.75	74.14

Table 6: The F1 score (%) of different base encoders.

higher than ours, indicating that the model trained by the explicit noise masking and the PHCE loss is more robust than the re-weighting strategy.

In Fig. 4, we choose the CoNLL-03 testing set and use t-SNE (Van der Maaten and Hinton 2008) tool to demonstrate the token representations in the semantic space. The model trained with Gaussian-based consistency regularization can make a clearer boundary between every two classes, corroborating our conclusions that avoiding the over-fitting problem and yielding better generalization capability.

Performance of Different Base Encoders

We end this section with a comparison with other base encoders in Table 6, including BERT+SoftMax, BERT+CRF, BiLSTM+SoftMax, and BiLSTM+CRF. Results show that our framework can be applied to arbitrary encoders. In addition, CRF is able to exploit the label dependency among the few data and further improves the overall performance of BERT and BiLSTM by 0.10% and 0.39%, respectively.

Conclusion

We propose a novel SeqUST framework for semi-supervised neural sequence labeling based on uncertainty-aware self-training. We utilize token-level model confidence and certainty to judiciously select reliable tokens in each unlabeled sentence. To mitigates the noisy labels, we introduce a masked sequence labeling task with noise-robust PHCE loss. We also present Gaussian-based consistency regularization to alleviate the over-fitting problem. Extensive experiments over multiple benchmarks show that our framework consistently outperforms strong baselines in low-resource settings. In the future, we will further improve the performance of denoising and apply our framework to other NLP tasks, such as extractive question answering, event extraction, etc.

Acknowledgements

References

Agarwal, O. 2022. Towards Robust Named Entity Recognition via Temporal Domain Adaptation and Entity Context Understanding. In *AAAI*, 12866–12867.

Amini, M.; Feofanov, V.; Pauletto, L.; Devijver, E.; and Maximov, Y. 2022. Self-Training: A Survey. *CoRR*, abs/2202.12040.

Andersen, J. S.; and Maalej, W. 2022. Efficient, Uncertainty-based Moderation of Neural Networks Text Classifiers. In *ACL*, 1536–1546.

Bengio, Y.; Louradour, J.; Collobert, R.; and Weston, J. 2009. Curriculum learning. In *ICML*, volume 382, 41–48.

Budzianowski, P.; Wen, T.; Tseng, B.; Casanueva, I.; Ultes, S.; Ramadan, O.; and Gasic, M. 2018. MultiWOZ - A Large-Scale Multi-Domain Wizard-of-Oz Dataset for Task-Oriented Dialogue Modelling. In *EMNLP*, 5016–5026.

Cao, P.; Zuo, X.; Chen, Y.; Liu, K.; Zhao, J.; and Bi, W. 2021. Uncertainty-Aware Self-Training for Semi-Supervised Event Temporal Relation Extraction. In *CIKM*, 2900–2904.

Chawla, N. V.; and Karakoulas, G. I. 2005. Learning From Labeled And Unlabeled Data: An Empirical Study Across Techniques And Domains. *JAIS*, 23: 331–366.

Chen, J.; Yang, Z.; and Yang, D. 2020. MixText: Linguistically-Informed Interpolation of Hidden Space for Semi-Supervised Text Classification. In *ACL*, 2147–2157.

Chen, L.; Ruan, W.; Liu, X.; and Lu, J. 2020. SeqVAT: Virtual Adversarial Training for Semi-Supervised Sequence Labeling. In *ACL*, 8801–8811.

Chen, M.; Tang, Q.; Livescu, K.; and Gimpel, K. 2018. Variational Sequential Labelers for Semi-Supervised Learning. In *EMNLP*, 215–226.

Clark, K.; Luong, M.; Manning, C. D.; and Le, Q. V. 2018. Semi-Supervised Sequence Modeling with Cross-View Training. In *EMNLP*, 1914–1925.

Coucke, A.; Saade, A.; Ball, A.; Bluche, T.; Caulier, A.; Leroy, D.; Doumouro, C.; Gisselbrecht, T.; Caltagirone, F.; Lavril, T.; Primet, M.; and Dureau, J. 2018. Snips Voice Platform: an embedded Spoken Language Understanding system for private-by-design voice interfaces. *CoRR*, abs/1805.10190.

Das, S. S. S.; Katiyar, A.; Passonneau, R. J.; and Zhang, R. 2022. CONTaiNER: Few-Shot Named Entity Recognition via Contrastive Learning. In *ACL*, 6338–6353.

Devlin, J.; Chang, M.; Lee, K.; and Toutanova, K. 2019. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In *NAACL*, 4171–4186.

Ding, B.; Liu, L.; Bing, L.; Kruengkrai, C.; Nguyen, T. H.; Joty, S. R.; Si, L.; and Miao, C. 2020. DAGA: Data Augmentation with a Generation Approach for Low-resource Tagging Tasks. *CoRR*, abs/2011.01549.

Do, K.; Tran, T.; and Venkatesh, S. 2021. Semi-Supervised Learning with Variational Bayesian Inference and Maximum Uncertainty Regularization. In *AAAI*, 7236–7244.

Gal, Y.; and Ghahramani, Z. 2016. Dropout as a Bayesian Approximation: Representing Model Uncertainty in Deep Learning. In *ICML*, volume 48, 1050–1059.

Gal, Y.; Islam, R.; and Ghahramani, Z. 2017. Deep Bayesian Active Learning with Image Data. In *ICML*, volume 70, 1183–1192.

Grandvalet, Y.; and Bengio, Y. 2004. Semi-supervised Learning by Entropy Minimization. In *NeurIPS*, 529–536.

Guo, C.; Pleiss, G.; Sun, Y.; and Weinberger, K. Q. 2017. On Calibration of Modern Neural Networks. In *ICML*, volume 70, 1321–1330.

Gururangan, S.; Dang, T.; Card, D.; and Smith, N. A. 2019. Variational Pretraining for Semi-supervised Text Classification. In *ACL*, 5880–5894.

Houlsby, N.; Huszar, F.; Ghahramani, Z.; and Lengyel, M. 2011. Bayesian Active Learning for Classification and Preference Learning. *CoRR*, abs/1112.5745.

Hu, Y.; and Khan, L. 2021. Uncertainty-Aware Reliable Text Classification. In *KDD*, 628–636.

Huang, J.; Li, C.; Subudhi, K.; Jose, D.; Balakrishnan, S.; Chen, W.; Peng, B.; Gao, J.; and Han, J. 2021. Few-Shot Named Entity Recognition: An Empirical Baseline Study. In *EMNLP*, 10408–10423.

Kim, H.; Son, J.; and Han, Y. 2022. LST: Lexicon-Guided Self-Training for Few-Shot Text Classification. *CoRR*, abs/2202.02566.

Kingma, D. P.; and Welling, M. 2014. Auto-Encoding Variational Bayes. In *ICLR*.

Kumar, M. P.; Packer, B.; and Koller, D. 2010. Self-Paced Learning for Latent Variable Models. In *NeurIPS*, 1189–1197.

Lafferty, J. D.; McCallum, A.; and Pereira, F. C. N. 2001. Conditional Random Fields: Probabilistic Models for Segmenting and Labeling Sequence Data. In *ICML*, 282–289.

Lee, S.; Kang, M.; Lee, J.; and Hwang, S. J. 2021. Learning to Perturb Word Embeddings for Out-of-distribution QA. In *ACL*, 5583–5595.

Li, C.; Li, X.; and Ouyang, J. 2021. Semi-Supervised Text Classification with Balanced Deep Representation Distributions. In *ACL*, 5044–5053.

Li, J.; Sun, A.; Han, J.; and Li, C. 2022. A Survey on Deep Learning for Named Entity Recognition. *IEEE TKDE*, 34(1): 50–70.

Li, X.; Feng, J.; Meng, Y.; Han, Q.; Wu, F.; and Li, J. 2020. A Unified MRC Framework for Named Entity Recognition. In *ACL*, 5849–5859.

Liu, J.; Pasupat, P.; Cyphers, S.; and Glass, J. R. 2013a. Asgard: A portable architecture for multilingual dialogue systems. In *ICASSP*, 8386–8390.

Liu, J.; Pasupat, P.; Wang, Y.; Cyphers, S.; and Glass, J. R. 2013b. Query understanding enhanced by hierarchical parsing structures. In *IEEE ASRU*, 72–77.

Liu, J.; Yu, M.; Chen, Y.; and Xu, J. 2022. Cross-Domain Slot Filling as Machine Reading Comprehension: A New Perspective. *IEEE ACM TASLP*, 30: 673–685.

Liu, Y.; Ma, C.; He, Z.; Kuo, C.; Chen, K.; Zhang, P.; Wu, B.; Kira, Z.; and Vajda, P. 2021. Unbiased Teacher for Semi-Supervised Object Detection. In *ICLR*.

Ma, T.; Jiang, H.; Wu, Q.; Zhao, T.; and Lin, C. 2022. Decomposed Meta-Learning for Few-Shot Named Entity Recognition. In *ACL*, 1584–1596.

Ma, X.; and Hovy, E. H. 2016. End-to-end Sequence Labeling via Bi-directional LSTM-CNNs-CRF. In *ACL*, 1064–1074.

Meng, Y.; Zhang, Y.; Huang, J.; Xiong, C.; Ji, H.; Zhang, C.; and Han, J. 2020. Text Classification Using Label Names Only: A Language Model Self-Training Approach. In *EMNLP*, 9006–9017.

Miller, S.; Guinness, J.; and Zamanian, A. 2004. Name Tagging with Word Clusters and Discriminative Training. In *NAACL*, 337–342.

Miyato, T.; Maeda, S.; Koyama, M.; and Ishii, S. 2019. Virtual Adversarial Training: A Regularization Method for Supervised and Semi-Supervised Learning. *IEEE TPAMI*, 41(8): 1979–1993.

Mukherjee, S.; and Awadallah, A. H. 2020. Uncertainty-aware Self-training for Few-shot Text Classification. In *NeurIPS*.

Peters, M. E.; Ammar, W.; Bhagavatula, C.; and Power, R. 2017. Semi-supervised sequence tagging with bidirectional language models. In *ACL*, 1756–1765.

Rizve, M. N.; Duarte, K.; Rawat, Y. S.; and Shah, M. 2021. In Defense of Pseudo-Labeling: An Uncertainty-Aware Pseudo-label Selection Framework for Semi-Supervised Learning. In *ICLR*.

Sang, E. F. T. K.; and Meulder, F. D. 2003. Introduction to the CoNLL-2003 Shared Task: Language-Independent Named Entity Recognition. In *NAACL*, 142–147.

Sennrich, R.; Haddow, B.; and Birch, A. 2016. Edinburgh Neural Machine Translation Systems for WMT 16. In *WMT*, 371–376.

Shen, Y.; Ma, X.; Tan, Z.; Zhang, S.; Wang, W.; and Lu, W. 2021. Locate and Label: A Two-stage Identifier for Nested Named Entity Recognition. In *ACL*, 2782–2794.

Snell, J.; Swersky, K.; and Zemel, R. S. 2017. Prototypical Networks for Few-shot Learning. In *NeurIPS*, 4077–4087.

Srivastava, N.; Hinton, G. E.; Krizhevsky, A.; Sutskever, I.; and Salakhutdinov, R. 2014. Dropout: a simple way to prevent neural networks from overfitting. *JMLR*, 15(1): 1929–1958.

Tsai, A. C.; Lin, S.; and Fu, L. 2022. Contrast-Enhanced Semi-supervised Text Classification with Few Labels. In *AAAI*, 11394–11402.

Van der Maaten, L.; and Hinton, G. 2008. Visualizing data using t-SNE. *Journal of machine learning research*, 9(11).

van Engelen, J. E.; and Hoos, H. H. 2020. A survey on semi-supervised learning. *Mach. Learn.*, 109(2): 373–440.

Wang, H.; and Yeung, D. 2016. Towards Bayesian Deep Learning: A Framework and Some Existing Methods. *IEEE TKDE*, 28(12): 3395–3408.

Wang, K.; Nie, Y.; Fang, C.; Han, C.; Wu, X.; Wang, X.; Lin, L.; Zhou, F.; and Li, G. 2022a. Double-Check Soft Teacher for Semi-Supervised Object Detection. In *IJCAI*, 1430–1436.

Wang, Y.; Mukherjee, S.; Chu, H.; Tu, Y.; Wu, M.; Gao, J.; and Awadallah, A. H. 2021a. Meta Self-training for Few-shot Neural Sequence Labeling. In *KDD*, 1737–1747.

Wang, Y.; Xu, C.; Sun, Q.; Hu, H.; Tao, C.; Geng, X.; and Jiang, D. 2022b. PromDA: Prompt-based Data Augmentation for Low-Resource NLU Tasks. In *ACL*, 4242–4255.

Wang, Z.; Li, Y.; Guo, Y.; and Wang, S. 2021b. Combating Noise: Semi-supervised Learning by Region Uncertainty Quantification. In *NeurIPS*, 9534–9545.

Weischedel, R.; Palmer, M.; Marcus, M.; Hovy, E.; Pradhan, S.; Ramshaw, L.; Xue, N.; Taylor, A.; Kaufman, J.; Franchini, M.; et al. 2013. Ontonotes release 5.0 ldc2013t19. *Linguistic Data Consortium*, 23.

Xie, Q.; Dai, Z.; Hovy, E. H.; Luong, T.; and Le, Q. 2020. Unsupervised Data Augmentation for Consistency Training. In *NeurIPS*.

Yang, X.; Song, Z.; King, I.; and Xu, Z. 2021. A Survey on Deep Semi-supervised Learning. *CoRR*, abs/2103.00550.

Yang, Z.; Hu, Z.; Salakhutdinov, R.; and Berg-Kirkpatrick, T. 2017. Improved Variational Autoencoders for Text Modeling using Dilated Convolutions. In *ICML*, volume 70, 3881–3890.

Yu, Y.; Kong, L.; Zhang, J.; Zhang, R.; and Zhang, C. 2021. ATM: An Uncertainty-aware Active Self-training Framework for Label-efficient Text Classification. *CoRR*, abs/2112.08787.

Zhang, X.; Yu, B.; Wang, Y.; Liu, T.; Su, T.; and Xu, H. 2022. Exploring Modular Task Decomposition in Cross-domain Named Entity Recognition. In *SIGIR*, 301–311.

Zhou, K.; Li, Y.; and Li, Q. 2022. Distantly Supervised Named Entity Recognition via Confidence-Based Multi-Class Positive and Unlabeled Learning. In *ACL*, 7198–7211.

Zhou, Q.; Yu, C.; Wang, Z.; Qian, Q.; and Li, H. 2021. Instant-Teaching: An End-to-End Semi-Supervised Object Detection Framework. In *CVPR*, 4081–4090.

Zhou, R.; Li, X.; He, R.; Bing, L.; Cambria, E.; Si, L.; and Miao, C. 2022. MELM: Data Augmentation with Masked Entity Language Modeling for Low-Resource NER. In *ACL*, 2251–2262.

Ziyadi, M.; Sun, Y.; Goswami, A.; Huang, J.; and Chen, W. 2020. Example-Based Named Entity Recognition. *CoRR*, abs/2008.10570.